INCREMENTAL PAGE TRANSITIONS ON ELECTRONIC PAPER DISPLAYS

BACKGROUND

A large and growing population of users is enjoying entertainment through the consumption of digital content, such as music, movies, images, electronic books, and so on. The users employ various electronic devices to consume such content. Among these electronic devices are electronic book (eBook) reader devices, cellular telephones, personal digital assistants (PDAs), portable media players, tablet computers, netbooks, and the like. As the quantity of available electronic media content continues to grow, along with increasing proliferation of devices to consume that media content, finding ways to enhance user experience continues to be a priority.

BRIEF DESCRIPTION OF THE DRAWINGS

The detailed description is set forth with reference to the accompanying figures. In the figures, the left-most digit(s) of a reference number identifies the figure in which the reference number first appears. The use of the same reference numbers in different figures indicates similar or identical items or features.

FIG. **1** illustrates an example electronic device that includes an electronic paper display and functionality for performing incremental page transitions on the electronic paper display.

FIGS. **2**A-**2**B illustrate an example flow diagram of the electronic device performing an incremental page transition in response to receiving a request from a user to proceed to a next page of a rendered content item.

FIG. **3** illustrates another example flow diagram of performing an incremental page transition when the electronic device implements a touch-sensitive electronic paper display. Here, the user requests to proceed to the next page of the rendered content item via a touch input and, in response, the device causes the page transition to track the user's touch input as the touch input moves across the touch-sensitive display.

FIG. **4** illustrates an example timeline of the device implementing an incremental page transition in response to receiving a request from a user to navigate from a first page of a content item to a second page. In this illustrated example, the device fills a framebuffer with pixel values corresponding to the second page and then incrementally updates the display with reference to the pixel values.

FIG. **5** illustrates another example timeline of the device implementing an incremental page transition in response to receiving the request from the user to navigate from the first page of the content item to the second page. In this illustrated example, the device incrementally fills the framebuffer with pixel values corresponding to the second page and then incrementally updates the display with reference to the pixel values corresponding to the second page.

FIG. **6** illustrates another example timeline of the device implementing an incremental page transition in response to receiving the request from the user to navigate from the first page of the content item to the second page. In this illustrated example, the device implements two framebuffers, the first of which includes the contents of the first page, which the device renders at a time, T**1**. In response to receiving the request, the device fills the second framebuffer with pixel values corresponding to the second page and then updates the display with reference to the first or second framebuffer depending upon a direction of the touch input across the display.

FIG. **7** illustrates an example process for performing an incremental page transition in response to a request from a user to navigate from a first page of a content item to a second page.

FIG. **8** illustrates an example process for performing an incremental page transition that tracks a touch input across a touch-sensitive electronic display.

FIG. **9** illustrates another example process for performing an incremental transition in response to a request from a user to navigate from a first portion of a content item to a second portion.

DETAILED DESCRIPTION

Overview

This disclosure describes, in part, techniques for implementing incremental page transitions on an electronic paper display of an electronic device. In some instances, an electronic-paper display controller that allows for per-pixel or multiple-pixel-block updating enables the incremental page transitions described herein.

Traditionally, when a user requests that an electronic device implementing an electronic paper display render a new piece of content, the device issues a single update request to a display controller of the device. That is, the device fills a framebuffer with a certain amount of pixel values (which may or may not correspond to the entire display) and then issues a single request to update the pixels associated with the pixel values in the framebuffer. In response, the display controller updates each pixel of the display (or each pixel of the block of pixels currently being updated) at approximately a same time. In contrast, the techniques described below may fill a framebuffer with a certain amount of pixel values and then issue sequential requests over a relatively short period of time, with each of the requests corresponding to a block of one or more pixels. As such, the techniques "uncover" the different portions of the pixels having values that have been filled in the framebuffer. Filling a framebuffer with pixel values and then sequentially updating those pixels in this manner allows for more aesthetically pleasing, animation-like updates as compared to traditional all-at-once updates.

To illustrate, envision that a user reading an electronic book an electronic device implementing an electronic paper display requests to turn from a first page of the electronic book to a second page. In response, the display controller may fill a framebuffer with a large block of pixel values and then initiate an update that includes a series of sequential requests that may be performed serially, overlapping with one another, or a combination thereof. To perform this update, the device may issue a series of requests over a predetermined schedule that uncovers the pixel values in the framebuffer over time. To issue the sequential requests, the device may send a separate instruction corresponding to each request to the controller, may send a single command including a schedule for the sequential requests, or may implement some combination thereof (e.g., sending instructions in groups of two). In either instance, the techniques first fill a framebuffer with values for a set of pixels and then sequentially update portions of that set of pixels over time, as opposed to simply filling a framebuffer, updating the entire framebuffer, and then repeating this process (as is done using traditional techniques).